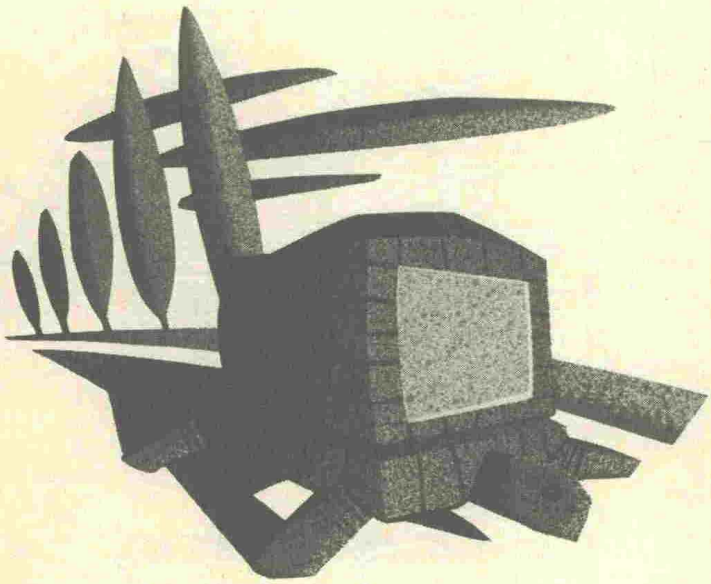
VISUAL BASIC

ขจร โรจนเมธินทร์

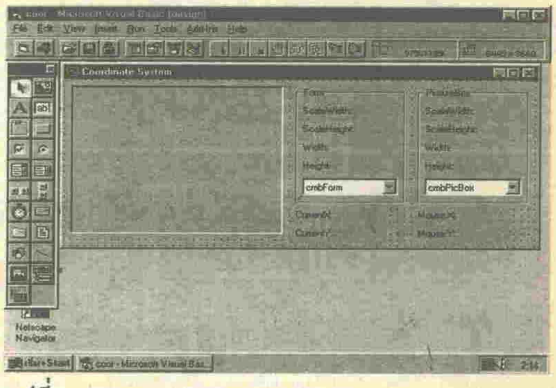
Visual BASIC : เรื่องของระบบโคออร์ดิเนต



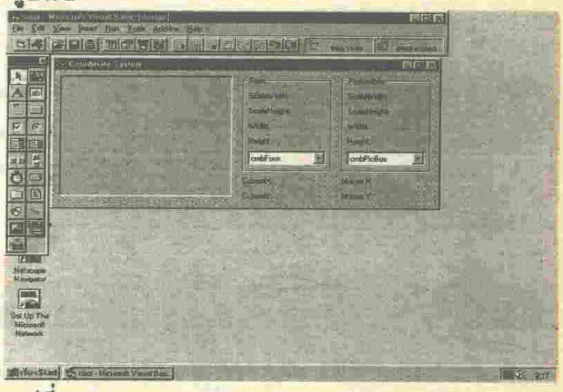
ระบบโคออร์ดิเนตและหน่วยวัดของวิชวลเบสิก เป็นความเข้าใจพื้นฐาน สำหรับใช้ในการเขียนจัดการกราฟิก

เนื่องจากวินโดวส์เป็นระบบปฏิบัติการที่สนับสนุนการใช้งานของการ์ดแสดงผลและจอภาพหลายชนิด ตั้งแต่ 640x350 พิกเซล (EGA), 640x480 (VGA), 800x600 (Super VGA), 1024x768 (Super VGA และ XGA) และอาจแสดงได้ละเอียดถึง 1280x1024 นอกจากความละเอียดที่แตกต่างกันแล้วยังสามารถแสดงจำนวนสีที่แตกต่างกันได้ตั้งแต่ 16, 256, 32,000, 65,000 หรือ 16 ล้านสี และเครื่องพิมพ์ได้หลากหลายชนิด เช่น ดอตแมตริกถึงโพสต์สคริปต์ การเขียนโปรแกรมที่จะจัดการกับเท็กซ์และกราฟิกให้ได้ตำแหน่งที่แน่นอนก็จะมีระบบโคออร์เนตที่เหมาะสมกับแต่ละงานไป

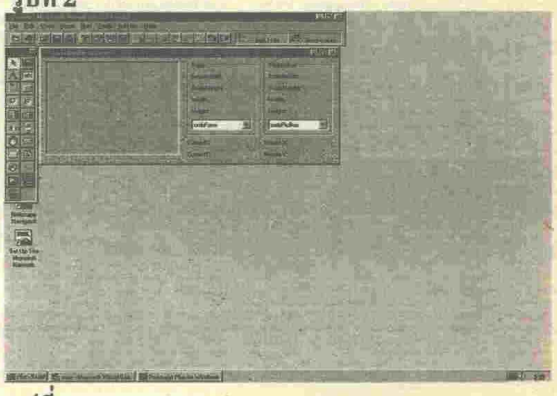
ความแตกต่างของความละเอียดจอภาพแต่ละประเภทจะทำให้ลักษณะของฟอร์มของที่ต่างกันไป โดยความละเอียดที่สูงขึ้นจะทำให้สามารถบรรจุส่วนประกอบต่าง ๆ ไว้ในจอภาพได้มากขึ้น จากรูปที่ 1, 2, 3 ซึ่งแสดงการเปลี่ยนแปลงที่เกิดขึ้นเมื่อผู้ใช้เลือกความละเอียดของจอภาพที่ต่างกัน ในแต่ละรูปนั้นมิได้เปลี่ยนแปลงขนาดฟอร์ม จะเห็นว่าเมื่อเลือกละเอียดมากขึ้นเราก็จะได้พื้นที่และคุณภาพในการแสดงมากขึ้น



รูปที่ 1



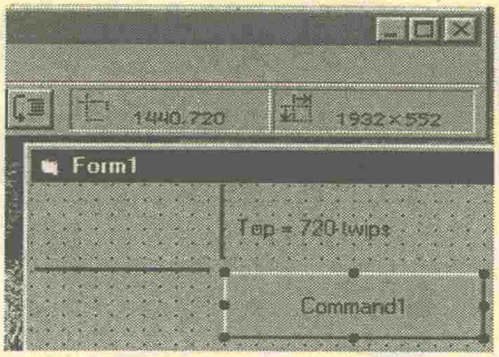
รูปที่ 2



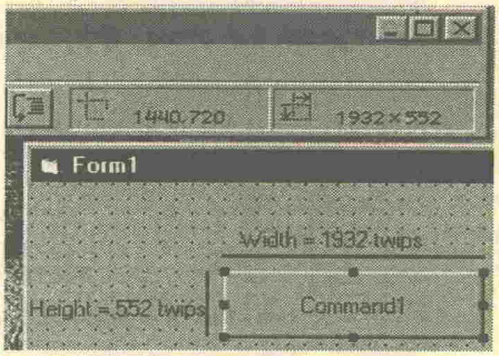
รูปที่ 3

หน่วยวัด	ความหมาย
Twip	1440 twips = 1 นิ้ว, 20 twips = 1 point
Point	72 points = 1 นิ้ว
Pixel	ขนาดเท่ากับพิกเซลของจอภาพ หรือจุดเล็กที่สุดที่เครื่องพิมพ์สามารถพิมพ์ได้
Character	แกน x : 120 twips = 1 character, แกน y : 240 twips = 1 character
Inch	1 inch = 1440 twips
Millimeter	254 millimeters = 1 inch, 5.67 twips = 1 millimeter
Centimeter	2.54 centimeters = 1 inch, 567 twips = 1 centimeter
User-defined	กำหนดโดยผู้เขียนโปรแกรม

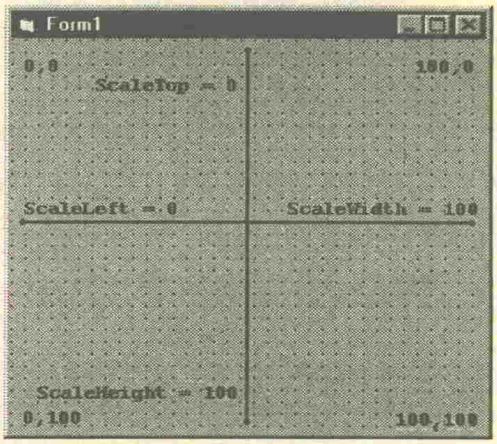
ตารางที่ 1 แสดงหน่วยวัดชนิดต่าง ๆ ที่สามารถกำหนดให้กับพรอพเพอร์ตี้ส์ ScaleMode



รูปที่ 4



รูปที่ 5



รูปที่ 6

ระบบหน่วยวัดของวิชวลเบสิก

เนื่องจากมีจอภาพชนิดที่แตกต่างกันมาก ดังนั้นจึงจำเป็นที่ระบบหน่วยวัดมาตรฐานขึ้นมาเพื่อความสะดวกในการเขียนโปรแกรมให้ไม่ติดอยู่กับอุปกรณ์ต่าง ๆ นี้ วิชวลเบสิกมีระบบหน่วยวัดของตัวเองหลายหน่วย ทั้งนี้เพื่อความสะดวกในการใช้งานประเภทต่าง ๆ เช่น งานด้านกราฟิกก็ต้องใช้หน่วยเป็นพิกเซล ในขณะที่งานด้านเวิร์ดโปรเซสซิงก็ควรใช้หน่วยเป็นพอนต์แทน กรณีที่ผู้ใช้ไม่กำหนดหน่วยวัดใหม่ วิชวลเบสิกจะใช้หน่วยโดยปริยายเป็น twips

การกำหนดหน่วยวัดที่จะใช้นั้นต้องกำหนดผ่านพรอพเพอร์ตี้ส์ ScaleMode (ดูตารางที่ 1) ออบเจ็กต์ที่มีใช้งานในวิชวลเบสิกและควรจะกำหนดค่าพรอพเพอร์ตี้ส์ ScaleMode ด้วย เพื่อให้ใช้งานได้ถูกต้อง เช่น ฟอร์มพิกเจอร์บ็อกซ์ พรินเตอร์ เป็นต้น

ระบบโคออร์ดิเนตของวิชวลเบสิก

ระบบโคออร์ดิเนตนั้นมีความจำเป็นในการเขียนโปรแกรมด้วยวิชวลเบสิกมากพอสมควร เริ่มตั้งแต่ช่วงที่ออกแบบฟอร์ม เนื่องจากเราจะต้องกำหนดตำแหน่งของออบเจ็กต์หรือคอนโทรลที่จะวางบนฟอร์ม โดยการกำหนดโคออร์ดิเนตของคอนโทรลเหล่านั้น พรอพเพอร์ตี้ส์ที่ใช้สำหรับกำหนดตำแหน่งของออบเจ็กต์ก็คือ Top และ Left ค่าโคออร์ดิเนตของพรอพเพอร์ตี้ส์ทั้งสองนี้จะใช้บอกว่าตำแหน่งของออบเจ็กต์นั้นอยู่ห่างจากมุมบนซ้ายของฟอร์ม พิกเจอร์บ็อกซ์ พรินเตอร์เท่าไร (ดูรูปที่ 4) จากรูปที่ 4 เรากำหนดตำแหน่งของคอมมานด์บัททอนให้อยู่ห่างจากมุมบนซ้ายของฟอร์มในแนวนอนเท่ากับ 1440 twips และห่างจากแนวดิ่งเท่ากับ 720 twips นอกจากนี้จะเห็นว่าทางด้านขวามือของทูลบาร์ของวิชวลเบสิกซึ่งแบ่งออกเป็น 2 ช่องนั้น ช่องทางซ้ายจะแสดงค่าของพรอพเพอร์ตี้ส์ Left และ Top ด้วย ขนาดของออบเจ็กต์แต่ละอันที่เราออกแบบสามารถกำหนดความกว้างและสูงได้ที่ค่าพรอพเพอร์ตี้ส์ Height และ Width (ดูรูปที่ 5) จะเห็นว่าช่องทางขวาสุดบนทูลบาร์ก็จะแสดงค่าของ Height และ Width ด้วยเช่นกัน

นอกจากพรอพเพอร์ตี้ส์ทั้ง 4 ที่กล่าวไปแล้ว ในวิชวลเบสิกยังมีพรอพเพอร์ตี้ส์ที่เกี่ยวข้องกับระบบโคออร์ดิเนตที่ควรทราบอีก 4 ชนิด คือ ScaleLeft, ScaleTop, ScaleWidth, ScaleHeight พรอพเพอร์ตี้ส์ชุดนี้ต่างจากชุดแรกที่พรอพเพอร์ตี้ส์ชุดหลังจะใช้กำหนดค่าที่อยู่ภายในฟอร์มเท่านั้น ไม่รวมขอบและไตเติ้ลบาร์ของฟอร์ม (ดูรูปที่ 6) การเปลี่ยนแปลงค่าพรอพเพอร์ตี้ส์ชุดหลังนี้จะไม่มีผลต่อขนาดและตำแหน่งของออบเจ็กต์ แต่จะผลต่อออบเจ็กต์ที่อยู่ภายในออบเจ็กต์นั้น

พรอพเพอร์ตี้ส์ CurrentX และ CurrentY

CurrentX และ CurrentY เป็นพรอพเพอร์ตี้ส์ที่เก็บค่าตำแหน่งที่อยู่ปัจจุบันในฟอร์ม พิกเจอร์บ็อกซ์ หรือพรินเตอร์ออบเจ็กต์ กรณีที่ผู้ใช้ไม่กำหนดชื่อออบเจ็กต์กับพรอพเพอร์ตี้ส์ วิชวลเบสิกจะถือว่าผู้ใช้ต้องการกำหนดค่าพรอเพอร์ตี้ส์ของฟอร์ม ค่าเริ่มต้นของพรอพเพอร์ตี้ส์ทั้งสองนี้จะถูกกำหนดให้เท่ากับค่าในพรอพเพอร์ตี้ส์ ScaleLeft, ScaleTop ของฟอร์ม พิกเจอร์บ็อกซ์ หรือพรินเตอร์ออบเจ็กต์ ค่าของ CurrentX และ CurrentY จะเปลี่ยนไปทุกครั้งที่ผู้ใช้วาดออบเจ็กต์หรือใช้เมธอดเหล่านี้ Circle, Cls, Line, Pset, Print, NewPage, EndDoc

```
รูปแบบการใช้ :
[form.]CurrentX [= x!]          [form.]CurrentY [= y!]
[form!]Name.CurrentX [= x!]  [form!]Name.CurrentY [= y!]
Printer.CurrentX [= x!]         Printer.CurrentY [= y!]
```

พรอพเพอร์ตี้ส์ Height และ Width

พรอพเพอร์ตี้ส์ Height และ Width ใช้กำหนดหรืออ่านค่าขนาดความสูงและกว้างของฟอร์ม คอนโทรล พรินเตอร์ หรือสกรีน สำหรับค่าของออบเจ็กต์ Printer และ Screen นั้นจะถูกกำหนดค่าโดยวินโดวส์และไม่สามารถกำหนดค่าได้ อ่านมาใช้ได้อย่างเดียว ดังนั้นเราจะทราบความละเอียดของจอภาพในแนวดิ่งหรือความสูงโดยอ่านค่าจากพรอพเพอร์ตี้ส์ Height ของ Screen ส่วนค่าของพรอพเพอร์ตี้ส์ Width ก็เป็นความกว้างของจอภาพ หากอ่านค่าของออบเจ็กต์ Printer ก็จะได้ขนาดความยาวและความกว้างของเพจที่พิมพ์ได้

```
รูปแบบการใช้ :
[form.] Height [= height!]    [form.]Width [ = w  idth!]
[form!]Name. Height [= height!] [form!]Name. Width [ = width!]
Printer. Height [= height!]   Printer. Width [ = width!]
Screen.Height [= height!]     Screen.Width [ = width!]
```

พรอพเพอร์ตี้ส์ Left และ Top

พรอพเพอร์ตี้ส์ทั้งสองนี้จะใช้กำหนดตำแหน่งของฟอร์มหรือคอนโทรลเทียบกับออบเจ็กต์ที่บรรจุอยู่ โดยปกติแล้วผู้ใช้ไม่ควรจะกำหนดพรอพเพอร์ตี้ส์ทั้งสองให้เกินไปจากความละเอียดที่จอภาพจะสามารถแสดงได้ พรอพเพอร์ตี้ส์ Left จะใช้กำหนดตำแหน่งในแนวนอน ส่วน Top เป็นตำแหน่งในแนวดิ่ง

```
รูปแบบการใช้ :
[form.]Left [= left!]      [form.]Top [= top!]
[form!]Name.Left [=left!] [form!]Name.Top [= top!]
```

พรอพเพอร์ตี้ส์ ScaleHeight, ScaleWidth, ScaleLeft และ ScaleTop

พรอพเพอร์ตี้ส์กลุ่มที่นำหน้าด้วย Scale นี้ จะมีความหมายคล้ายกับกลุ่มที่ไม่มีคำว่า Scale นำ ต่างกันเพียงในกลุ่มนี้จะไม่นับเอาขอบและไตเติลเข้ามารวมด้วย เช่น ถ้าเป็นฟอร์ม ScaleHeight ก็คือความสูงภายในฟอร์มเท่านั้นไม่รวมขอบ ไตเติลบาร์และเมนูบาร์ ส่วน ScaleWidth จะเป็นความกว้างภายในฟอร์มไม่รวมขนาดขอบ สำหรับ ScaleLeft และ ScaleTop นั้นเป็นค่าที่มุมบนซ้ายในฟอร์ม ซึ่งปกติจะมีค่าเป็น 0 เสมอ

พรอพเพอร์ตี้ส์ ScaleMode

เราจะกำหนดหน่วยวัดโดยการกำหนดค่าตามตารางที่ 1 ในพรอพเพอร์ตี้ส์ ScaleMode ไปในตอนต้นแล้ว การกำหนดค่า ScaleMode ของออบเจ็กต์ใดก็ตามจะมีผลต่อค่า ScaleWidth และ ScaleHeight ของออบเจ็กต์นั้น ส่วนค่า Height และ Width จะไม่เปลี่ยนแปลง เนื่องจากพรอพเพอร์ตี้ส์จะเปลี่ยนแปลงค่า ScaleMode ของออบเจ็กต์อันใหญ่ที่ห่อหุ้มออบเจ็กต์นั้นไว้ เช่น ถ้าเราเปลี่ยนแปลงค่า ScaleMode ของฟอร์มก็จะมีผลต่อค่า Height และ Width ของคอนโทรลที่บรรจุอยู่ภายในฟอร์มนั้น ส่วนถ้าเราเปลี่ยนแปลงค่า ScaleMode ของคอนโทรลก็จะมีผลต่อค่า ScaleHeight และ ScaleWidth ของคอนโทรลเท่านั้น

เมธอด Scale

เมธอด Scale นี้ใช้สำหรับกำหนดค่าพรอพเพอร์ตี้ส์ ScaleHeight, ScaleWidth, ScaleTop และ ScaleLeft พร้อมกันด้วยคำสั่งเดียว

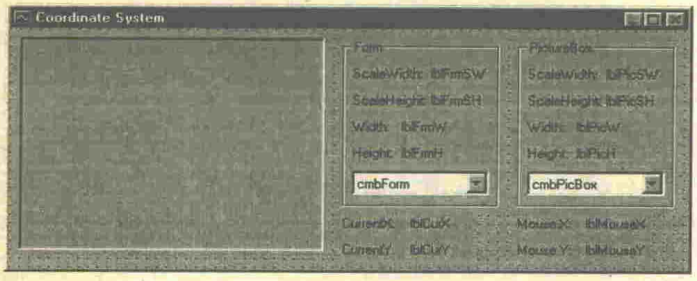
```
รูปแบบการใช้ :
      [form.]Scale [(x1!, y1!) - (x2!, y2!)]
      [form!]Name.Scale [(x1!, y1!) - (x2!, y2!)]
      Printer.Scale [(x1!, y1!) - (x2!, y2!)]
```

(x1!, y1!) เป็นค่าที่จะกำหนดให้พรอพเพอร์ตี้ส์ ScaleLeft และ ScaleTop

(x2!, y2!) เป็นค่าที่จะกำหนดให้พรอพเพอร์ตี้ส์ ScaleWidth และ ScaleHeight

ตัวอย่างโปรแกรม

ตัวอย่างโปรแกรมที่ผู้เขียนใช้ในบทความนี้เป็นโปรแกรมง่าย ๆ ที่จะแสดงค่าพรอพเพอร์ตี้ส์ที่เกี่ยวข้องกับระบบโคออร์ดิเนต ได้แก่ ScaleWidth, ScaleHeight, Width และ Height โดยแยกเป็น 2 กลุ่ม คือ พรอพเพอร์ตี้ส์ของกลุ่มฟอร์มและของพิกเจอร์บ็อกซ์ (ดูรูปที่ 7) เราจัดให้พรอพเพอร์ตี้ส์อยู่รวมกันภายในคอนโทรลเฟรม 2 อัน นอกจากนี้



รูปที่ 7

ในคอนโทรลเฟรมทั้งสองจะมีคอมโบบ็อกซ์สำหรับใช้กำหนดค่าให้พรอพเพอร์ตี้ส์ ScaleMode ของฟอร์มและพิกเจอร์บ็อกซ์ตามลำดับ ซึ่งเมื่อมีการเปลี่ยนแปลงค่า ScaleMode ผลที่เกิดขึ้นกับพรอพเพอร์ตี้ส์ทั้ง 4 อันก็จะเปลี่ยนแปลงตามไปด้วย นอกจากนี้โปรแกรมยังจะแสดงค่าตำแหน่งขณะที่ผู้ใช้เลื่อนเม้าส์ไปมาบนพิกเจอร์บ็อกซ์และแสดงค่าตำแหน่งปัจจุบันของโคออร์ดิเนตของพิกเจอร์บ็อกซ์ด้วย กรณีที่ผู้ใช้คลิ๊กเม้าส์บนพิกเจอร์บ็อกซ์แล้วลากเม้าส์ไปมาโปรแกรมจะวาดเส้นตรงระหว่างจุดแรกและจุดสุดท้ายที่เลื่อนเม้าส์ไปให้เห็น ซึ่งขณะวาดเส้นตรงนี้ ผู้อ่านจะเห็นความแตกต่างค่าโคออร์ดิเนตที่แสดงหลังเลเบล CurrentX, CurrentY, Mouse X และ Mouse Y ได้

ก่อนอื่นจะขอกล่าวถึงขั้นตอนการออกแบบฟอร์ม จะเห็นว่าเราใช้คอนโทรลเลเบลเกือบทั้งหมดในการแสดงค่าของพรอพเพอร์ตี้ส์ คอนโทรลเหล่านี้แบ่งเป็น 2 กลุ่มใหญ่ คือกลุ่มที่ต้องแก้ไขพรอพเพอร์ตี้ส์ Caption ส่วนอีกกลุ่มจะแก้ไขพรอพเพอร์ตี้ส์ Name ให้ขึ้นต้นด้วย "lbl..." แล้วตามด้วย "Pic..." หรือ "Frm..." ก็ขึ้นกับว่าเป็นคอนโทรลในกลุ่มเฟรม PictureBox หรือ Form ส่วนชื่อเลเบลก็ลงท้ายด้วย "SW", "SH" ก็เป็นกลุ่ม ScaleWidth, ScaleHeight หรือ "W", "H" ก็เป็นกลุ่ม Width และ Height (ดูตารางที่ 2)

ออบเจ็กต์	พรอพเพอร์ตี้ส์	ค่าเริ่มต้น
VB.Form	Name	Form1
	Caption	"Coordinate System"
VB.Frame	Name	Frame2
	Caption	"PictureBox"
VB.ComboBox	Name	cmbPicBox
	Style	2 'Dropdown List
VB.Label	Name	lblPicH
VB.Label	Name	lblPicW
VB.Label	Name	lblPicSH
VB.Label	Name	lblPicSW
VB.Label	Name	Label8
	Caption	"Height:"
VB.Label	Name	Label7
	Caption	"Width:"
VB.Label	Name	Label6
	Caption	"ScaleHeight:"
VB.Label	Name	Label5
	Caption	"ScaleWidth:"
VB.Frame	Name	Frame1
	Caption	"Form"
VB.ComboBox	Name	cmbForm
	Style	2 'Dropdown List
VB.Label	Name	lblFrmH
VB.Label	Name	lblFrmW
VB.Label	Name	lblFrmSH
VB.Label	Name	lblFrmSW
VB.Label	Name	Label4
	Caption	"Height:"
VB.Label	Name	Label3
	Caption	"Width:"
VB.Label	Name	Label2
	Caption	"ScaleHeight:"
VB.Label	Name	Label1
	Caption	"ScaleWidth:"
VB.PictureBox	Name	Picture1
VB.Label	Name	lblMouseY
VB.Label	Name	lblMouseX
VB.Label	Name	lblCurY
VB.Label	Name	lblCurX
VB.Label	Name	Label12
	Caption	"Mouse Y:"
VB.Label	Name	Label11
	Caption	"Mouse X:"
VB.Label	Name	Label10
	Caption	"CurrentY:"
VB.Label	Name	Label9
	Caption	"CurrentX:"

ตารางที่ 2 แสดงส่วนประกอบที่ต้องออกแบบบนฟอร์มหลักของโปรแกรม

ลิสติงที่ 1 แสดงโค้ดทั้งหมดของโปรแกรมตัวอย่าง

```
Dim drawing As Integer
Dim startX As Single
Dim startY As Single
Dim oldX As Single
Dim oldY As Single

Private Sub cmbForm_Click()
    ScaleMode = cmbForm.ListIndex
      lblCurX.Caption = Format(Picture1.CurrentX, "####0.00")
      lblCurY.Caption = Format(Picture1.CurrentY, "####0.00")
      lblFrmSW.Caption = Format(ScaleWidth, "####0.00")
      lblFrmSH.Caption = Format(ScaleHeight, "####0.00")
      lblPicW.Caption = Format(Picture1.Width, "####0.00")
      lblPicH.Caption = Format(Picture1.Height, "####0.00")
End Sub

Private Sub cmbPicBox_Click()
      Picture1.ScaleMode = cmbPicBox.ListIndex
      lblCurX.Caption = Format(Picture1.CurrentX, "####0.00")
      lblCurY.Caption = Format(Picture1.CurrentY, "####0.00")
      lblPicSW.Caption = Format(Picture1.ScaleWidth, "####0.00")
      lblPicSH.Caption = Format(Picture1.ScaleHeight, "####0.00")
End Sub

Private Sub Form_Load()
      lblCurX.Caption = Format(Picture1.CurrentX, "####0.00")
      lblCurY.Caption = Format(Picture1.CurrentY, "####0.00")

      lblFrmSW.Caption = Format(ScaleWidth, "####0.00")
      lblFrmSH.Caption = Format(ScaleHeight, "####0.00")
      lblFrmW.Caption = Format(Width, "####0.00")
      lblFrmH.Caption = Format(Height, "####0.00")

      lblPicSW.Caption = Format(Picture1.ScaleWidth, "####0.00")
      lblPicSH.Caption = Format(Picture1.ScaleHeight, "####0.00")
      lblPicW.Caption = Format(Picture1.Width, "####0.00")
      lblPicH.Caption = Format(Picture1.Height, "####0.00")

      cmbForm.AddItem "Twips"
      cmbForm.AddItem "Points"
      cmbForm.AddItem "Pixels"
      cmbForm.AddItem "Characters"
      cmbForm.AddItem "Inches"
      cmbForm.AddItem "Millimeters"
      cmbForm.AddItem "Centimeters"
      cmbForm.ListIndex = 0

      cmbPicBox.AddItem "Twips"
      cmbPicBox.AddItem "Points"
      cmbPicBox.AddItem "Pixels"
      cmbPicBox.AddItem "Characters"
      cmbPicBox.AddItem "Inches"
      cmbPicBox.AddItem "Millimeters"
      cmbPicBox.AddItem "Centimeters"
      cmbPicBox.ListIndex = 0

      Top = (Screen.Width - Width) \ 2
      Left = (Screen.Height - Height) \ 2
End Sub

Private Sub Form_Resize()
      lblFrmSW.Caption = Format(ScaleWidth, "####0.00")
      lblFrmSH.Caption = Format(ScaleHeight, "####0.00")
      lblFrmW.Caption = Format(Width, "####0.00")
      lblFrmH.Caption = Format(Height, "####0.00")
End Sub
```

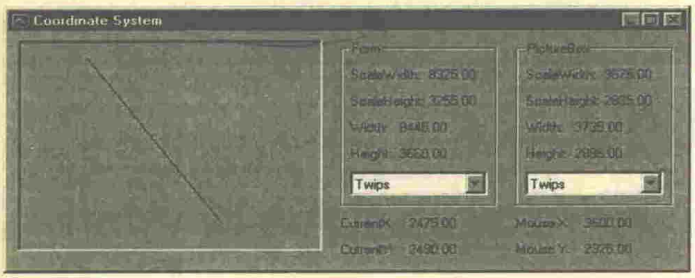
→

ลิสติงที่ 1 (ต่อ)

```
Private Sub Picture1_MouseDown(Button As Integer, Shift As Integer, X As Single, Y As Single)
    Picture1.PSet (X, Y)
    startX = X
    startY = Y
    oldX = X
    oldY = Y
    drawing = True
    lblCurX.Caption = Format(Picture1.CurrentX, "####0.00")
    lblCurY.Caption = Format(Picture1.CurrentY, "####0.00")
End Sub

Private Sub Picture1_MouseMove(Button As Integer, Shift As Integer, X As Single, Y As Single)
    If drawing Then
       Picture1.DrawWidth = 1
       If (oldX <> X) Or (oldY <> Y) Then Picture1.Cls
       Picture1.Line (startX, startY)-(X, Y)
    End If
    lblMouseX.Caption = Format(X, "####0.00")
    lblMouseY.Caption = Format(Y, "####0.00")
End Sub

Private Sub Picture1_MouseUp(Button As Integer, Shift As Integer, X As Single, Y As Single)
    drawing = False
    lblCurX.Caption = Format(Picture1.CurrentX, "####0.00")
    lblCurY.Caption = Format(Picture1.CurrentY, "####0.00")
End Sub
```



รูปที่ 8

ซับรูทีน cmbForm_Click และ cmbPicBox_Click ทำหน้าที่เปลี่ยนแปลงค่าของ ScaleMode ตามแต่กรณี และแสดงผลที่เกิดกับพรอพเพอร์ตี้ส์ทั้งสี่ตัวเมื่อมีการเปลี่ยนแปลงค่า ScaleMode, ซับรูทีน Picture1_MouseDown จะกำหนดตำแหน่งโคออร์ดิเนตปัจจุบันใหม่ด้วย Pset เก็บค่าโคออร์ดิเนตขณะนั้นไว้เป็นจุดเริ่มต้นสำหรับวาดเส้นตรง และสุดท้ายก็แสดงตำแหน่งขณะที่ผู้ใช้เลื่อนเม้าส์ที่เลเบล lblCurX และ lblCurY, ซับรูทีน Picture1_MouseMove มีหน้าที่หลักคือวาดเส้นตรงด้วยคำสั่ง Line (ดูรูปที่ 8) ในระหว่างที่ตัวแปร drawing เป็นจริง โดยเมื่อผู้ใช้ขยับเม้าส์ค่า X,Y ก็จะเปลี่ยนไปไม่เท่ากับ oldX และ oldY ก็จะต้องลบเส้นตรงเส้นเดิมที่วาดไว้ก่อนจะวาดเส้นใหม่ด้วย สำหรับซับรูทีน Picture1_MouseUp นั้นจะกำหนดค่าตัวแปร drawing ให้ไม่จริงเพื่อแสดงว่าเลิกวาดเส้นตรงแล้ว

บทความนี้จะช่วยให้ผู้อ่านเข้าใจระบบโคออร์ดิเนตและหน่วยวัดของวิชวลเบสิกได้ดีขึ้น เนื่องจากในการเขียนจัดการกราฟิกนั้นเราจะต้องมีความเข้าใจพื้นฐานเรื่องนี้พอสมควร ซึ่งเมื่อเข้าใจดีแล้วก็จะช่วยให้เขียนโปรแกรมได้สนุกขึ้นมาก